

# **BLOCK RAM GENERATOR USER-Guide Version 2**

Project by:

MERL Organization

# Table of Contents

1: Cloning Block-RAM-Generator.....	3
2: Requirements.....	3
3: Getting Started with Block-RAM-Generator.....	3
3.1 Launch GUI.....	3
3.2 Block RAM Model.....	4
3.3 Output Specification.....	5
3.4 Generating Files.....	6
3.4.1 Model Same as Base Width.....	6
3.4.2 Model Greater than Base Width.....	7
3.4.3 RTL Wrapper.....	8
3.4.4 Simulation File.....	8
3.4.5 FPGA Synthesis Script.....	9
3.4.6 Multiple Files.....	10
3.5 Compile and Generate.....	10
3.5.1 Compile.....	10
3.5.2 Specification Block Diagrams.....	11
3.5.2.1 Model Same as Base Width.....	11
3.5.2.1.1 1R1W Memory.....	11
3.5.2.1.2 NR1W Memory.....	12
3.5.2.1.3 NRNW Memory.....	13
3.5.2.2 Model Greater Than Base Width.....	13
3.5.2.2.1 1R1W Memory.....	13
3.5.2.2.2 NR1W Memory.....	14
3.5.2.2.3 NRNW Memory.....	14
3.5.3 Generate.....	15
3.6 Files Location.....	15
3.7 Simulation and Verification.....	16
3.7.1 Automate.....	16
3.7.2 Manual.....	17
3.8 FPGA Synthesis.....	18

# 1: Cloning Block-RAM-Generator

Clone the Git repo from the given URL:

<https://github.com/merledu/RAM-Generator.git>

Command to clone the repo

```
git clone https://github.com/merledu/RAM-Generator.git
```

## 2: Requirements

Before using the generator make sure, you have the following, tools install on your computer/server.

- 1) PyQT 5
- 2) Python 3.5 or higher.
- 3) Icarus Verilog version 11
- 4) GTK Wave
- 5) Vivado 2018.2
- 6) Graphviz

## 3: Getting Started with Block-RAM-Generator

### 3.1 Launch GUI

Run the BlockRAM.py script by executing the following command

```
pyhton3 /Path/to/RAM-Generator/compiler/BlockRAM.py
```

The GUI interface window will appear.

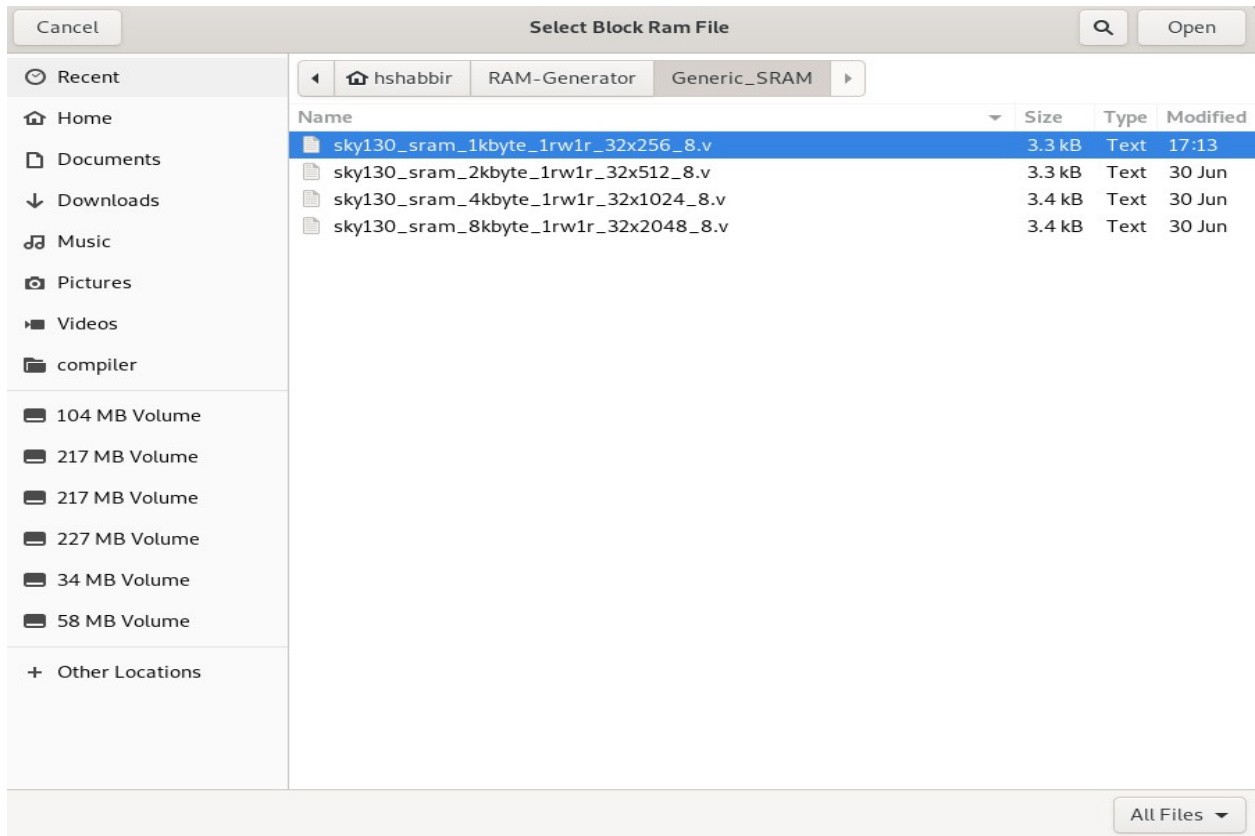
The screenshot shows the 'Block RAM Generator' GUI window. It has a title bar 'Block RAM Generator' and a close button. The main area is divided into several sections:

- Block Ram:** Contains input fields for 'Model File Path' (with a 'browse' button), 'Number of Read ports', 'Number of Write ports', 'Number of Address ports', 'Memory depth', 'Data Width (Bits)', 'Address Width (Bits)', and a 'Fixed Width' checkbox.
- Output Specification:** Contains input fields for 'Number of Read ports', 'Number of Write ports', 'Number of Address ports', 'Memory depth', 'Data Width (Bits)', and 'Address Width (Bits)'. It also has checkboxes for 'ASIC Synthesis script', 'FPGA Synthesis script', 'RTL wrapper', and 'Simulation script'.
- Output Memory:** Contains input fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', 'Data Width (Bits)', 'Address Width (Bits)', 'Number of Banks', 'Number of Block Rams', and 'Output Files Path'.
- Banks:** Contains input fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', and 'Address Width (Bits)'.

At the bottom, there are buttons for 'Simulate', 'Verify', 'Compile', and 'Generate'.

### 3.2 Block RAM Model

Select the SRAM base model as an input Block RAM provided in **RAM-Generator/Generic\_SRAM**.



After the selection of the input file for the Block RAM, provide the information for the remaining inputs. As we are using SRAM as a Block RAM, so the remaining inputs will be.

Block Ram

Model File Path

/home/waleeds1/Mer1/Block-RAM-Generator/Generic\_SRAM/sram.v

browse

Number of Read ports

1

Number of Write ports

1

Number of Address ports

1

Memory depth

32

Data Width (Bits)

32

Address Width (Bits)

5

Fixed Width

☐

Output Specification

Number of Read ports

Number of Write ports

Number of Address ports

Memory depth

Data Width (Bits)

Address Width (Bits)

ASIC Synthesis script

☐

FPGA Synthesis script

☐

RTL wrapper

☐

Simulation script

☐

Output Memory

Number of Read ports

Number of Write ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Number of Banks

Number of Block Rams

Output Files Path

Banks

Number of Read ports

Number of Write ports

Memory depth

Address Width (Bits)

Compile

Generate

Input the information regarding the specification you want in **Output Specification** column.

### Case 1: 1R1W

Block Ram

Model File Path
/home/waleeds1/Mer1/Block-RAM-Generator/Generic\_SRAM/sram.v
browse

Number of Read ports

Number of Write ports

Number of Address ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Fixed Width
☐

Output Specification

Number of Read ports

Number of Write ports

Number of Address ports

Memory depth

Data Width (Bits)

Address Width (Bits)

ASIC Synthesis script
☐

FPGA Synthesis script
☐

RTL wrapper
☐

Simulation script
☐

Output Memory

Number of Read ports

Number of Write ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Number of Banks

Number of Block Rams

Output Files Path

Banks

Number of Read ports

Number of Write ports

Memory depth

Address Width (Bits)

Compile

Generate

## Case 2: NR1W

Block RAM Generator

Block Ram

Model File Path  
/home/waleeds1/Mer1/Block-RAM-Generator/Generics\_SRAM/sram.v browse

Number of Read ports  
1

Number of Write ports  
1

Number of Address ports  
1

Memory depth  
32

Data Width (Bits)  
32

Address Width (Bits)  
5

Fixed Width ☐

Output Specification

Number of Read ports  
5

Number of Write ports  
1

Number of Address ports

Memory depth  
64

Data Width (Bits)  
32

Address Width (Bits)  
6

ASIC Synthesis script ☐

FPGA Synthesis script ☐

RTL wrapper ☐

Simulation script ☐

Output Memory

Number of Read ports

Number of Write ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Number of Banks

Number of Block Rams

Output Files Path

Banks

Number of Read ports

Number of Write ports

Memory depth

Address Width (Bits)

Compile

Generate

### Case 3: NRNW

The screenshot shows the 'Block RAM Generator' application window. The interface is divided into three main sections: 'Block Ram', 'Output Specification', and 'Output Memory'. The 'Block Ram' section on the left contains input fields for 'Model File Path' (set to a file path), 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (32), 'Data Width (Bits)' (32), 'Address Width (Bits)' (5), and a 'Fixed Width' checkbox which is unchecked. The 'Output Specification' section in the middle contains 'Number of Read ports' (2), 'Number of Write ports' (3), 'Number of Address ports' (1), 'Memory depth' (64), 'Data Width (Bits)' (32), 'Address Width (Bits)' (6), and checkboxes for 'ASIC Synthesis script', 'FPGA Synthesis script', 'RTL wrapper', and 'Simulation script', all of which are unchecked. The 'Output Memory' section on the right contains input fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', 'Data Width (Bits)', 'Address Width (Bits)', 'Number of Banks', 'Number of Block Rams', and 'Output Files Path'. Below this section is a 'Banks' sub-section with 'Number of Read ports', 'Number of Write ports', 'Memory depth', and 'Address Width (Bits)'. At the bottom right of the window are 'Compile' and 'Generate' buttons.

## 3.4 Generating Files

### 3.4.1 Model Same as Base Width

To generate the model with same width as of the base model width do not check the **Fixed width** check box.

Block RAM Generator

Block RAM Generator

Block Ram

Model File Path4-Generator/Generic\_SRAM/sky130\_sram\_1kbyte\_1rw1r\_32x256\_8.vbrowse

Number of Read ports1

Number of Write ports1

Number of Address ports1

Memory depth256

Data Width (Bits)32

Address Width (Bits)8

Fixed Width☐

Output Specification

Number of Read ports1

Number of Write ports1

Number of Address ports

Memory depth512

Data Width (Bits)32

Address Width (Bits)9

ASIC Synthesis script☐

FPGA Synthesis script☐

RTL wrapper☐

Simulation script☐

Output Memory

Number of Read ports

Number of Write ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Number of Banks

Number of Block Rams

Output Files Path

Banks

Number of Read ports

Number of Write ports

Memory depth

Address Width (Bits)

SimulateVerify

CompileGenerate

### 3.4.2 Model Greater than Base Width

If you want to generate the model with greater width than the base model select the **Fixed Width** check box.

**Note:** You can only generate the models with the greater width in the multiples of base model width.

Block RAM Generator

Block RAM Generator

Block Ram

Model File Path4-Generator/Generic\_SRAM/sky130\_sram\_1kbyte\_1rw1r\_32x256\_8.vbrowse

Number of Read ports1

Number of Write ports1

Number of Address ports1

Memory depth256

Data Width (Bits)32

Address Width (Bits)8

Fixed Width☒

Output Specification

Number of Read ports1

Number of Write ports1

Number of Address ports

Memory depth512

Data Width (Bits)128

Address Width (Bits)9

ASIC Synthesis script☐

FPGA Synthesis script☐

RTL wrapper☐

Simulation script☐

Output Memory

Number of Read ports

Number of Write ports

Memory depth

Data Width (Bits)

Address Width (Bits)

Number of Banks

Number of Block Rams

Output Files Path

Banks

Number of Read ports

Number of Write ports

Memory depth

Address Width (Bits)

SimulateVerify

CompileGenerate

### 3.4.3 RTL Wrapper

To generate the RTL wrapper select the RTL wrapper check box. The generator will only generate the RTL file.

The screenshot shows the 'Block RAM Generator' window. The 'Block Ram' tab is active, displaying fields for 'Model File Path' (set to /home/waleeds1/Me1/Block-RAM-Generator/Generic\_SRAM/sram.v), 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (32), 'Data Width (Bits)' (32), 'Address Width (Bits)' (5), and 'Fixed Width' (unchecked). The 'Output Specification' tab is also visible, showing 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (64), 'Data Width (Bits)' (32), 'Address Width (Bits)' (6), and checkboxes for 'ASIC Synthesis script' (unchecked), 'FPGA Synthesis script' (unchecked), 'RTL wrapper' (checked), and 'Simulation script' (unchecked). The 'Output Memory' tab is also visible, showing fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', 'Data Width (Bits)', 'Address Width (Bits)', 'Number of Banks', 'Number of Block Rams', and 'Output Files Path'. The 'Banks' tab is also visible, showing fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', and 'Address Width (Bits)'. The 'Compile' and 'Generate' buttons are at the bottom right.

### 3.4.4 Simulation File

To generate the simulation file select the Simulation script check box. The generator will generate the test-bench with the Make File.

The screenshot shows the 'Block RAM Generator' window. The 'Block Ram' tab is active, displaying fields for 'Model File Path' (set to /home/waleeds1/Me1/Block-RAM-Generator/Generic\_SRAM/sram.v), 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (32), 'Data Width (Bits)' (32), 'Address Width (Bits)' (5), and 'Fixed Width' (unchecked). The 'Output Specification' tab is also visible, showing 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (64), 'Data Width (Bits)' (32), 'Address Width (Bits)' (6), and checkboxes for 'ASIC Synthesis script' (unchecked), 'FPGA Synthesis script' (unchecked), 'RTL wrapper' (unchecked), and 'Simulation script' (checked). The 'Output Memory' tab is also visible, showing fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', 'Data Width (Bits)', 'Address Width (Bits)', 'Number of Banks', 'Number of Block Rams', and 'Output Files Path'. The 'Banks' tab is also visible, showing fields for 'Number of Read ports', 'Number of Write ports', 'Memory depth', and 'Address Width (Bits)'. The 'Compile' and 'Generate' buttons are at the bottom right.



### 3.4.5 FPGA Synthesis Script

To generate the FPGA synthesis script select the FPGA Synthesis script check box. A pop up dialog box will appear asking for the FPGA device name. Input the FPGA device name. The generator will generate the **synth.tcl** file.

The screenshot shows the 'Block RAM Generator' window with the following settings:

- Block Ram:** Model File Path: /home/waleeds1/Mer1/Block-RAM-Generator/Generic\_SRAM/sram.v (browse); Number of Read ports: 1; Number of Write ports: 1; Number of Address ports: 1; Memory depth: 32; Data Width (Bits): 32; Address Width (Bits): 5; Fixed Width: ☐.
- Output Specification:** Number of Read ports: 1; Number of Write ports: 1; Number of Address ports: (disabled); Memory depth: 64; Data Width (Bits): 32; Address Width (Bits): 6; ASIC Synthesis script: ☐; **FPGA Synthesis script: ☒**; RTL wrapper: ☐; Simulation script: ☐.
- Output Memory:** Number of Read ports: (empty); Number of Write ports: (empty); Memory depth: (empty); Data Width (Bits): (empty); Address Width (Bits): (empty); Number of Banks: (empty); Number of Block Rams: (empty); Output Files Path: (empty).
- Banks:** Number of Read ports: (empty); Number of Write ports: (empty); Memory depth: (empty); Address Width (Bits): (empty).

Buttons at the bottom: **Compile** (blue), **Generate** (grey).

This screenshot is identical to the previous one, but with an 'FPGA name' dialog box open in the center. The dialog box contains the text 'Enter FPGA name:' followed by a text input field. At the bottom of the dialog are two buttons: 'Cancel' (red X) and 'OK' (green checkmark).

The example of FPGA device name is **xc7a35ticsg324-1L**

### 3.4.6 Multiple Files

To generate the multiple files select the multiple check box as per your requirement. The generator will generate the files according to it.

The screenshot shows the 'Block RAM Generator' window. It is divided into three main sections: 'Block Ram', 'Output Specification', and 'Output Memory'.  
- **Block Ram:** Includes fields for 'Model File Path' (set to a local path), 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (1), 'Memory depth' (32), 'Data Width (Bits)' (32), 'Address Width (Bits)' (5), and a 'Fixed Width' checkbox.  
- **Output Specification:** Includes 'Number of Read ports' (1), 'Number of Write ports' (1), 'Number of Address ports' (empty), 'Memory depth' (64), 'Data Width (Bits)' (32), 'Address Width (Bits)' (6), and checkboxes for 'ASIC Synthesis script' (unchecked), 'FPGA Synthesis script' (checked), 'RTL wrapper' (checked), and 'Simulation script' (checked).  
- **Output Memory:** Includes 'Number of Read ports' (empty), 'Number of Write ports' (empty), 'Memory depth' (empty), 'Data Width (Bits)' (empty), 'Address Width (Bits)' (empty), 'Number of Banks' (empty), 'Number of Block Rams' (empty), and 'Output Files Path' (empty).  
At the bottom right, there are 'Compile' and 'Generate' buttons.

## 3.5 Compile and Generate

### 3.5.1 Compile

Press the **Compile** button to see the result of your desire specification.

This screenshot shows the same 'Block RAM Generator' window after the 'Compile' button has been pressed. The configuration is identical to the previous state, but the 'Output Files Path' field in the 'Output Memory' section now contains the path: 'erator/compiler/output/sram64x32\_1rw\_2021-03-26\_23-06-11'. The 'Compile' and 'Generate' buttons remain at the bottom right.

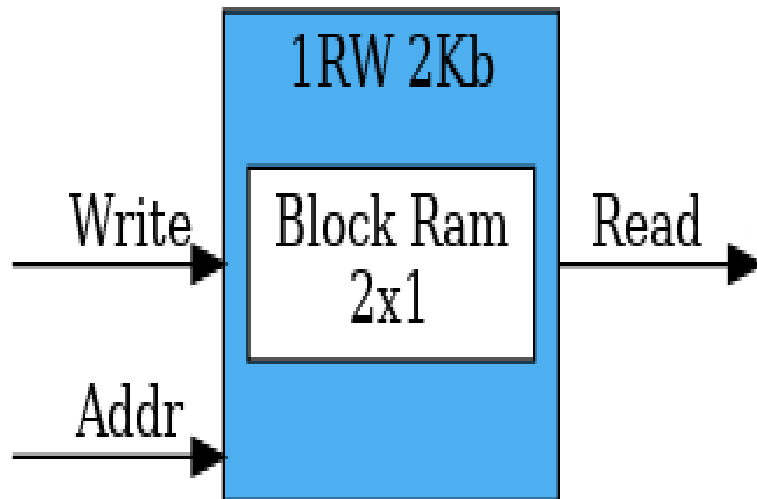
If you want to change your specification, repeat 3.3, 3.4, and 3.5.1.

### 3.5.2 Specification Block Diagrams

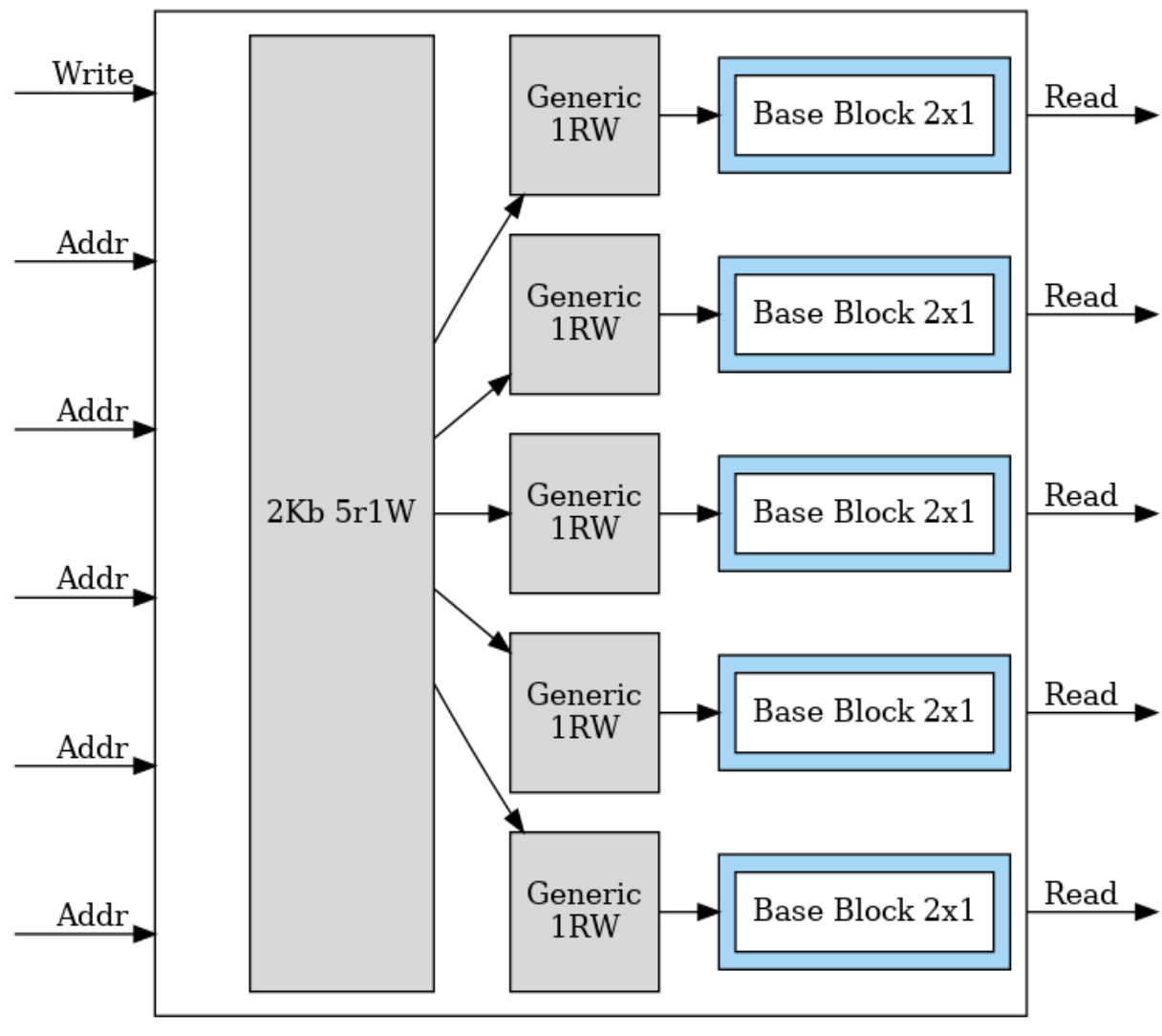
When you press the **Compile** button the block diagram of your desire specification pop up.

#### 3.5.2.1 Model Same as Base Width

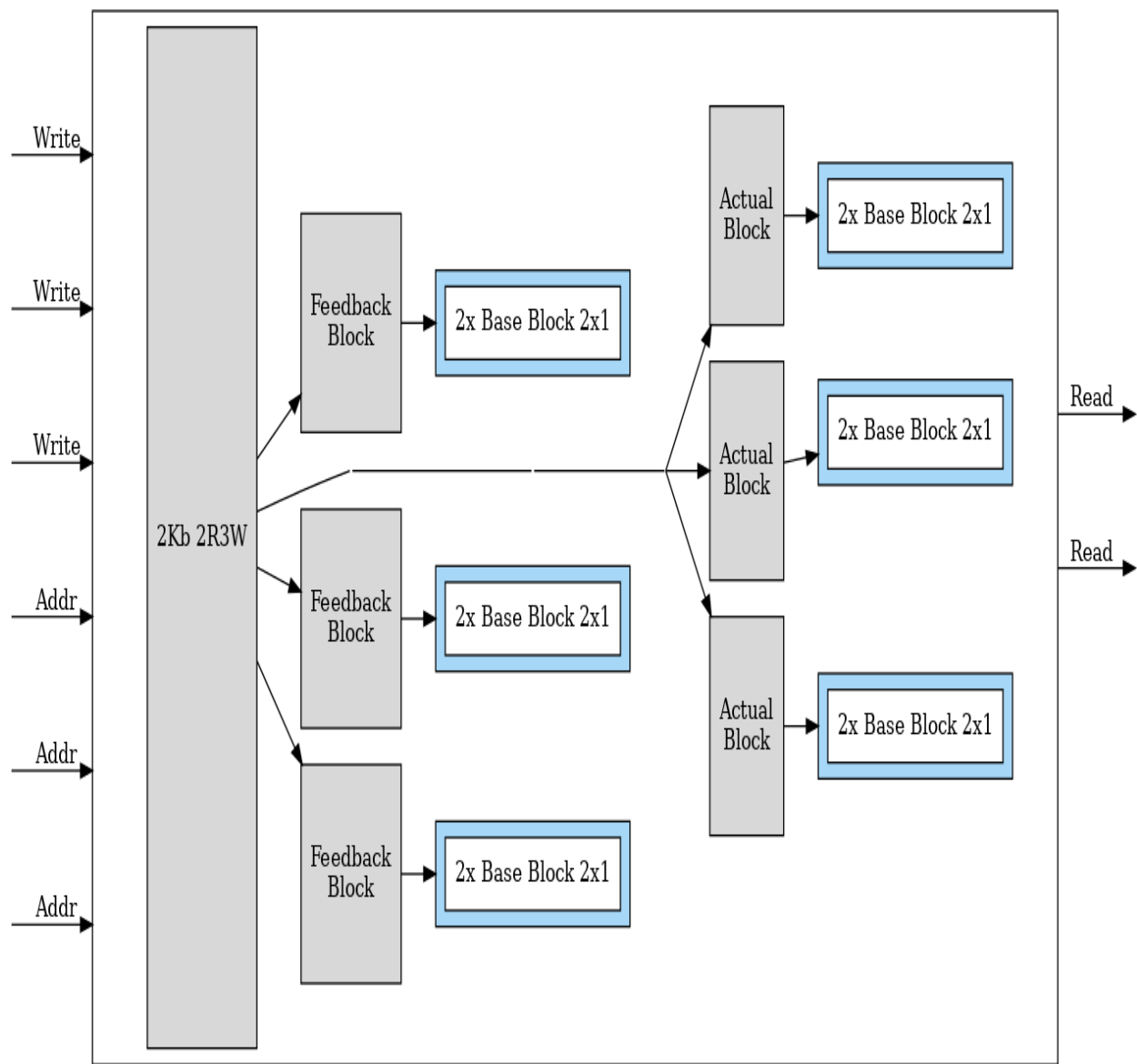
##### 3.5.2.1.1 1R1W Memory



### 3.5.2.1.2 NR1W Memory

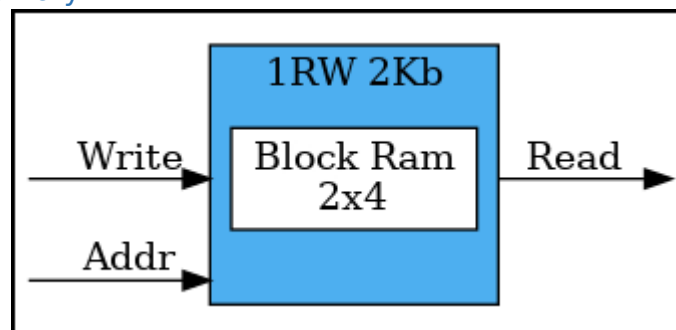


### 3.5.2.1.3 NRNW Memory

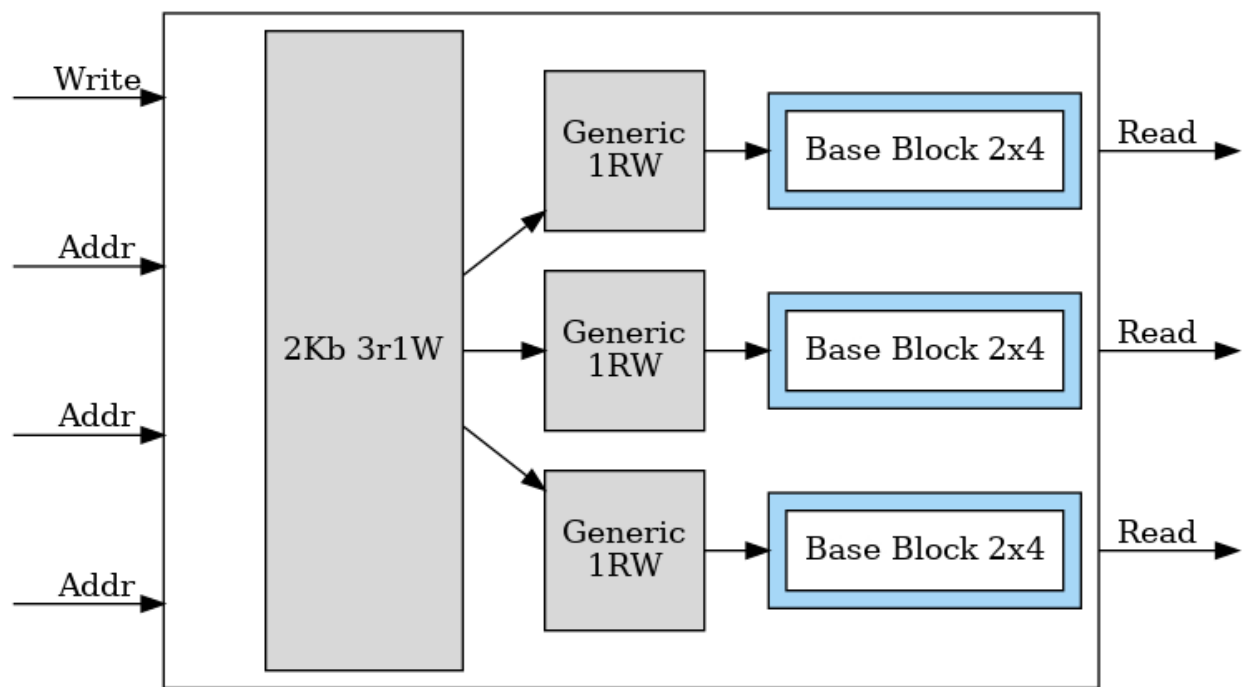


### 3.5.2.2 Model Greater Than Base Width

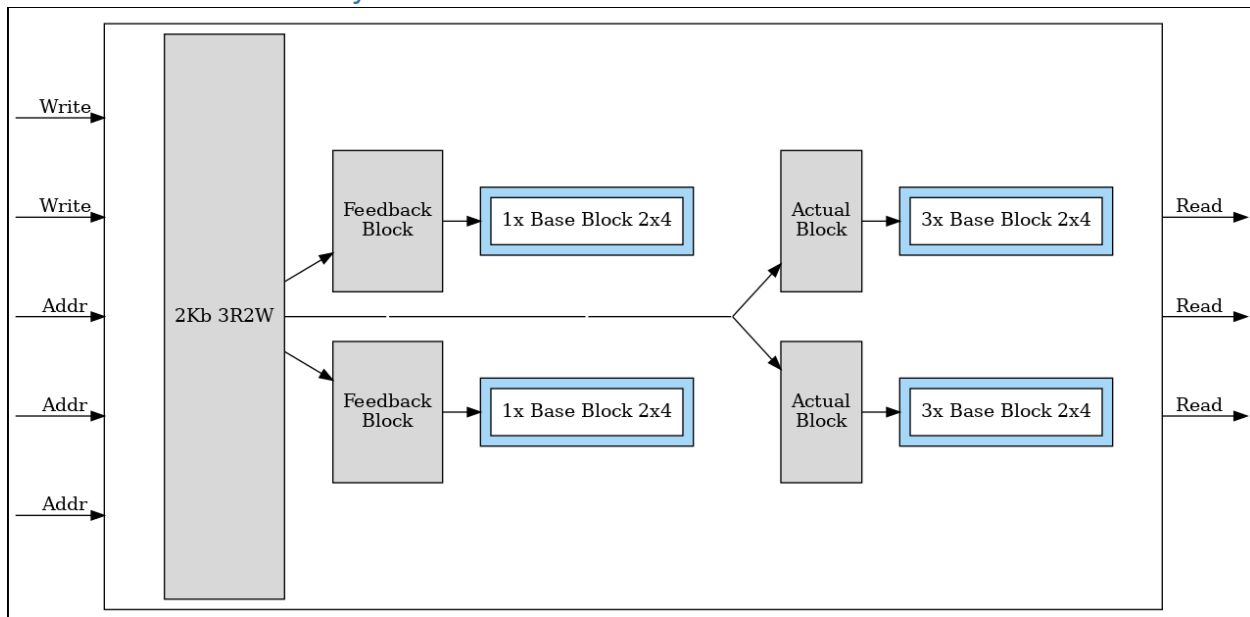
#### 3.5.2.2.1 1R1W Memory



### 3.5.2.2.2 NR1W Memory

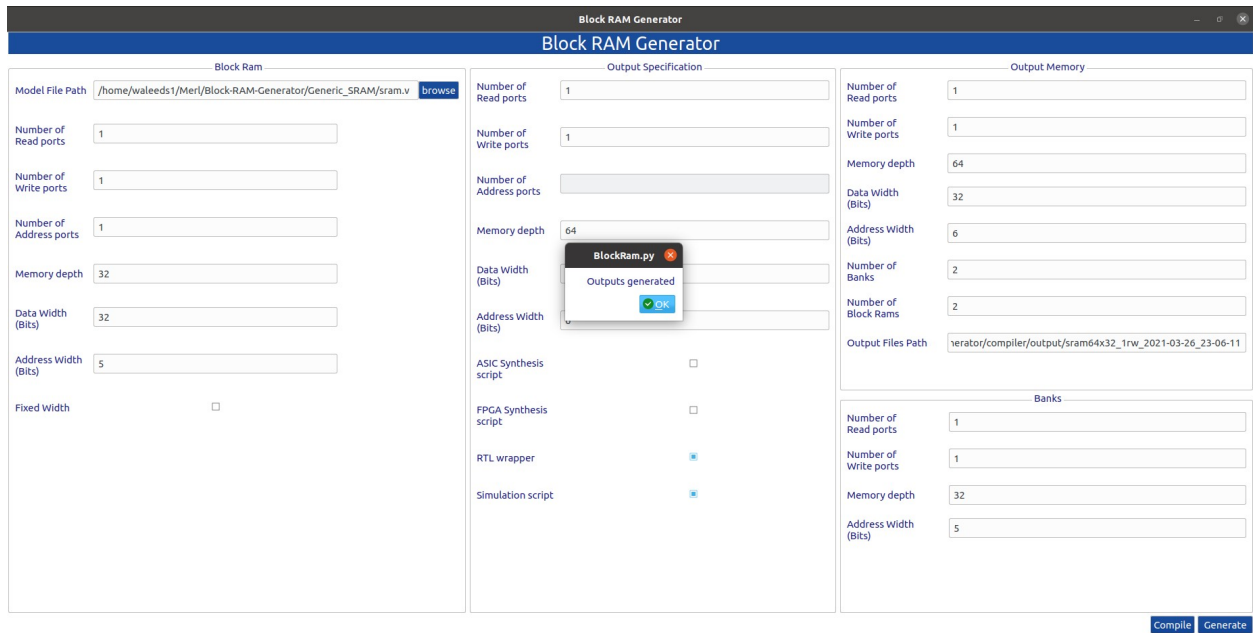


### 3.5.2.2.3 NNRW Memory



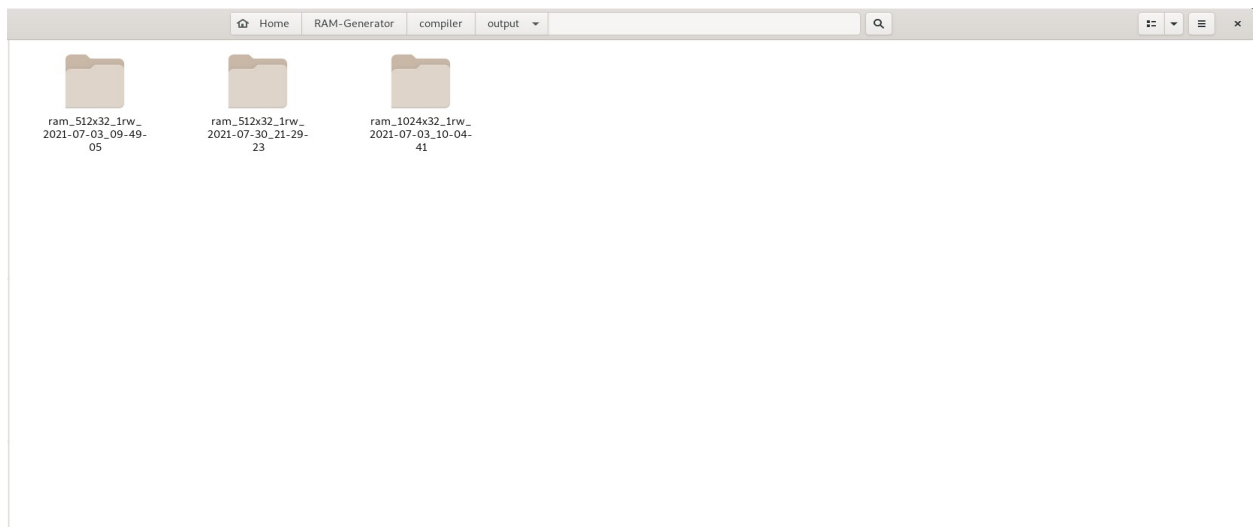
### 3.5.3 Generate

Press the **Generate** button to generate your files, a pop up message will occur indicating that your files are generated.



### 3.6 Files Location

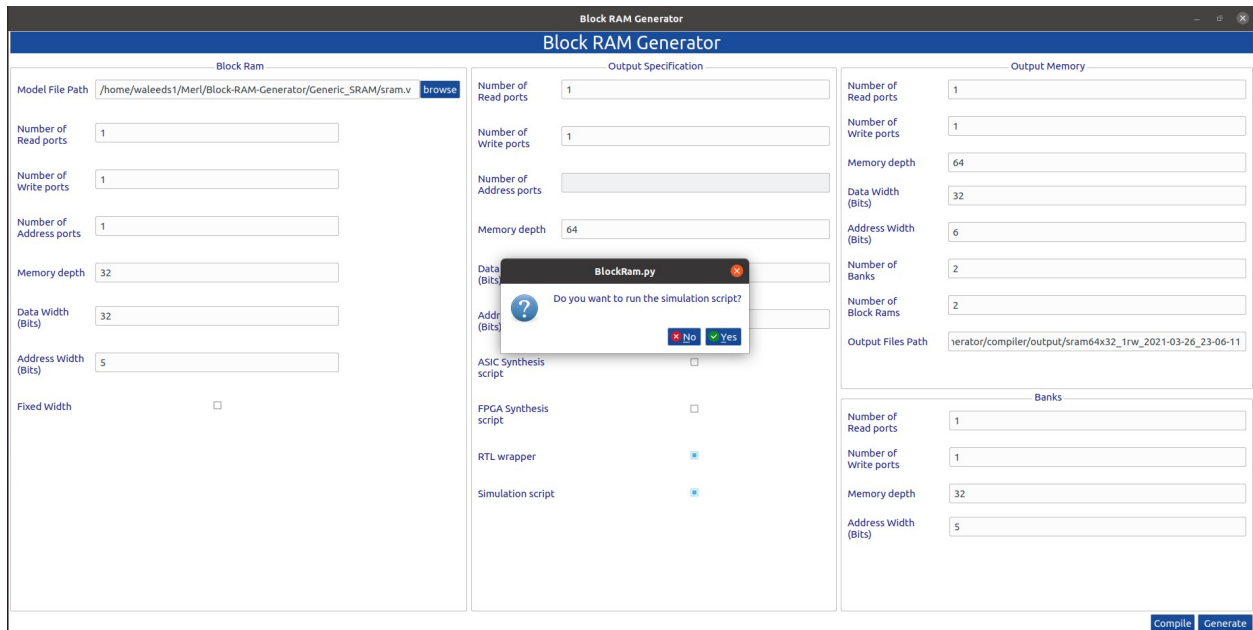
To view your generated files go to **RAM-Generator/compiler/output** folder.



## 3.7 Simulation and Verification

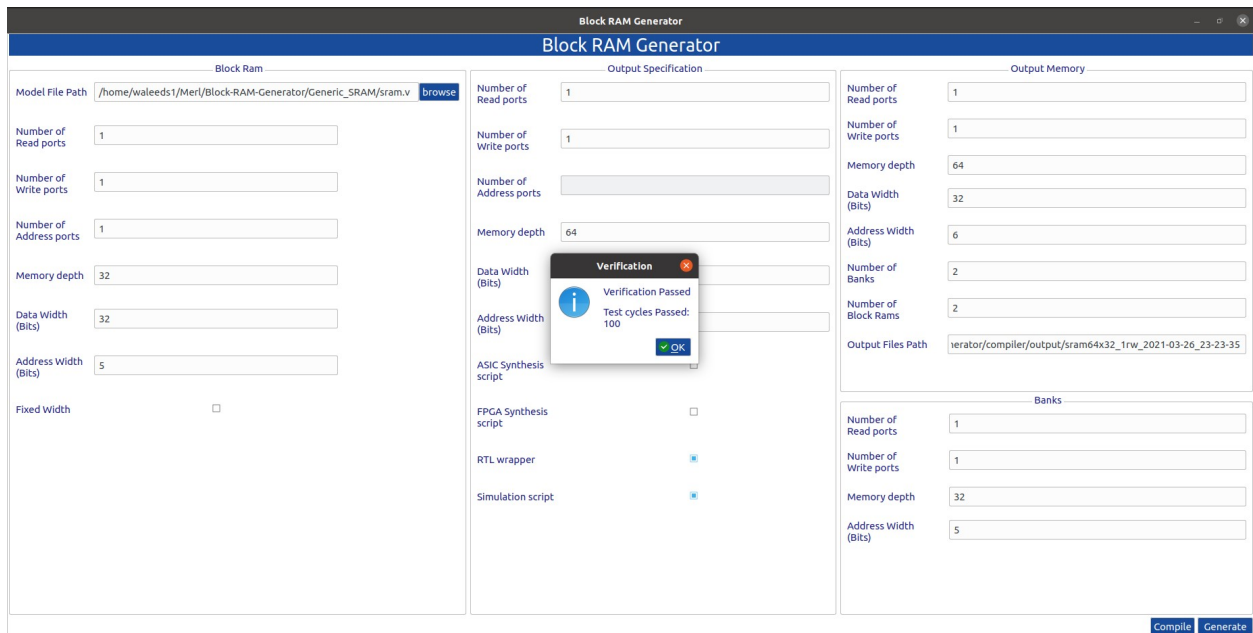
### 3.7.1 Automate

A pop up message will occur after the file generation asking if you want to simulate and verify the design automatically or not.



Press **yes**, if you want to automate the process.

A pop up message will occur showing that your design is pass or not.





### 3.7.2 Manual

To simulate the design manually follow the following steps:

- 1) Go to the directory where the files are generated.

```
cd Path/to/RAM-Generator/compiler/output/ram_generated_(user_specification)/
```

- 2) If you have select, the check box to generate the simulation script a **Makefile** will be available in the output folder.



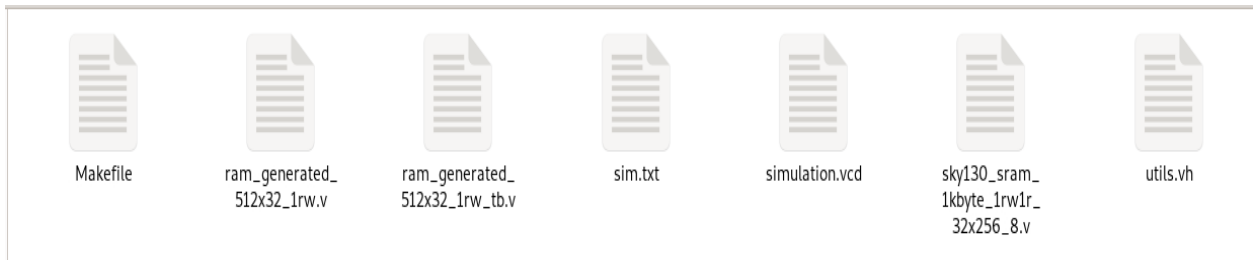
- 3) Open the terminal and run **make** command.

```
~/RAM-Generator/compiler/output/ram_512x32_1rw_2021-07-30_21-29-23$ make
```

- 4) Will get the following output at terminal.

```
VCD info: dumpfile simulation.vcd opened for output.
Simulating SRAM:
Write ports : 1
Read ports  : 1
Data width  : 32
RAM depth   : 64
Address width: 6
```

- 5) Simulation and log file will be generate.



- 6) To view the waveform open the **simulation.vcd** file.
- 7) To view the log file open the **sim.txt** file.

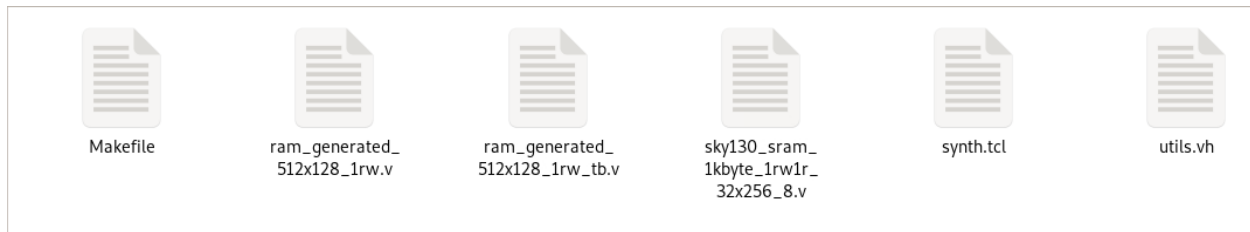
## 3.8 FPGA Synthesis

To synthesize the design using the tcl file follow the following steps.

- 1) Go to the directory where the files are generated.

```
cd Path/to/RAM-Generator/compiler/output/ram_generated_(user_specification)/
```

- 2) If you have select, the check box to generate the FPGA synthesis script a **synth.tcl** file will be available in the output folder.



- 3) To execute the **synth.tcl** file from terminal, run the following command.

```
vivado -mode batch -source synth.tcl
```

**Note: Run the terminal command from the same folder mention in step 1.**

- 4) To execute the **synth.tcl** file from Vivado, run the following commands.
  - 1) Open Vivado.
  - 2) Go to directory where the files are generated.
  - 3) In Tcl Console type **source synth.tcl** and press enter.

